**<https://www.microsoft.com/en-us/learning/dashboard.aspx>**

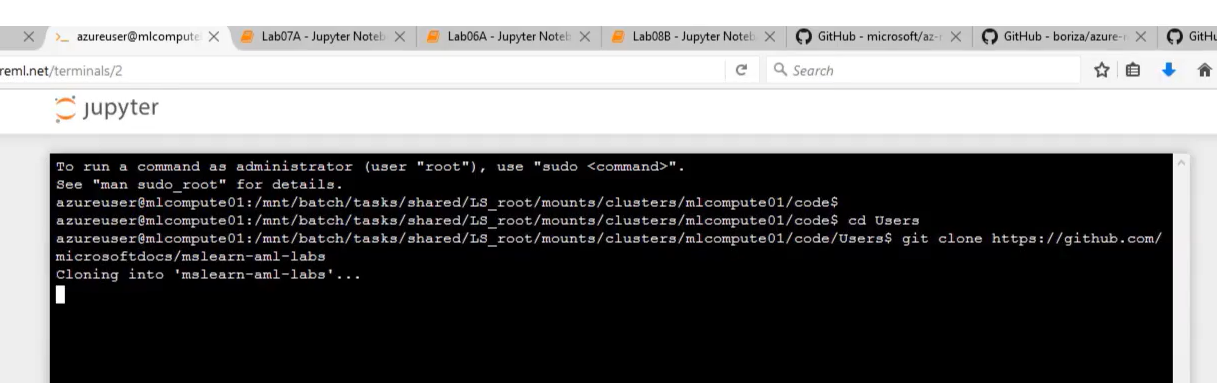
Hands on guide for labs:

https://tinyurl.com/amlvhol

To clone these labs:

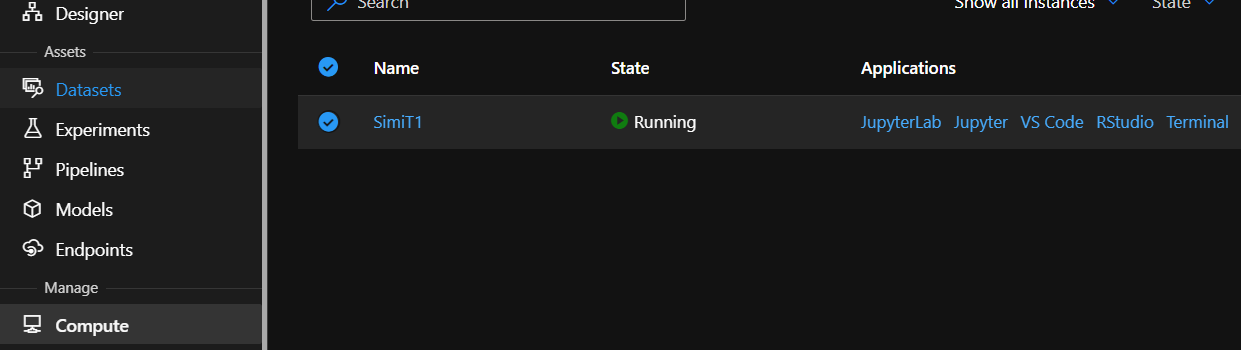
In the Jupyter notebook Go to folder Users start Terminal :

(In the webinar instructions are in the last 5 minutes)

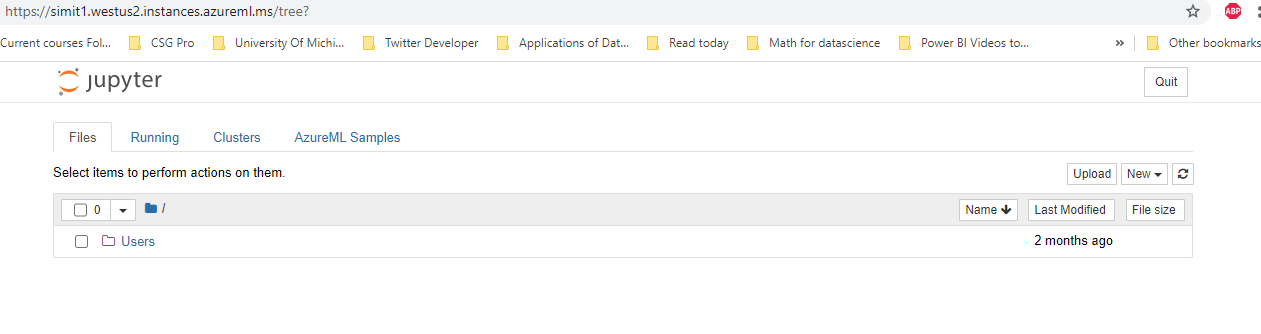


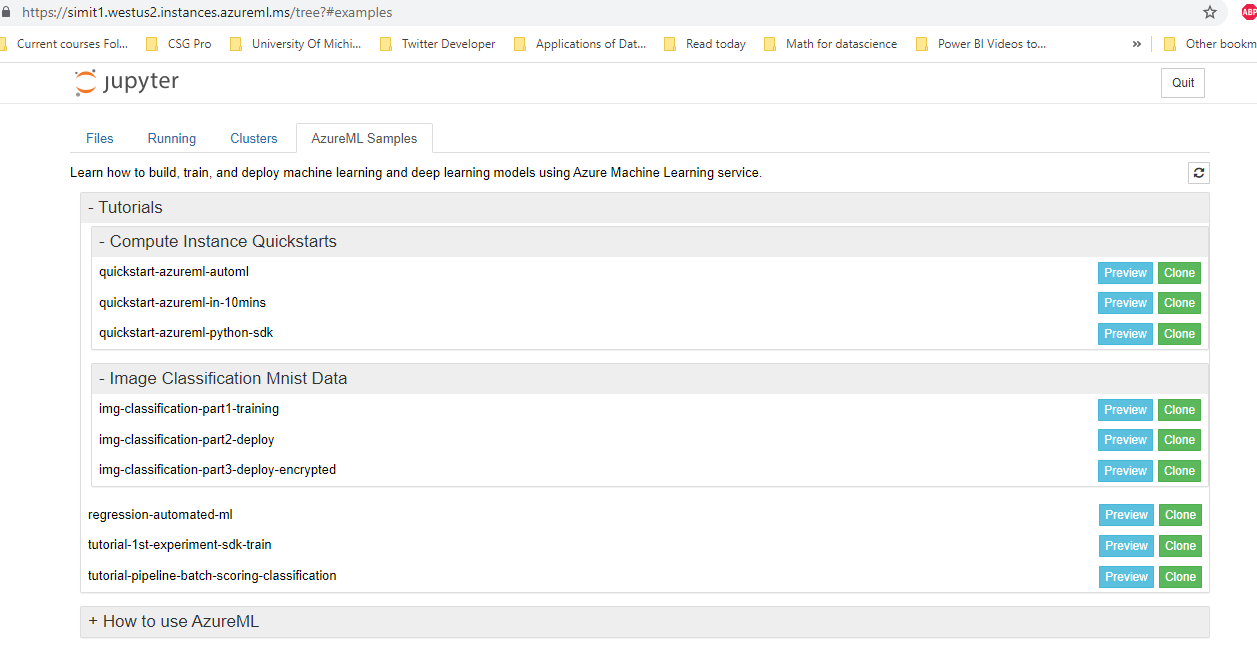
**To get to example code:**

1. **In AzureML studio – click on Compute:**



1. **Click on Jupyter**



1. 
2. **Make sure AKS is for production ONLY**
3. **When have a deployment InferenceConfig set up, you have a AciWebservice – but you need a deployment CONDA file. (Just deploy may not mean production-local, ACI,AKS)**
4. **Clusters – SCALING Azure Databricks cluster – Attached compute. Compute Clusters (CPU) low priority and hence not production worthy)**
5. **Best model – run.get\_output**
6. **For Real Time streaming – IoT and Azure Stack Edge**
7. **To stream logs and monitor Run Status – pipeline\_run.wait\_for\_completion()**
8. **PipelineRunStep (batch inferencing) - init() and run(mini\_batch)**
9. **Model deployment failure – Azure Monitor**
10. **Show output files of a run get\_file\_names()**
11. **Registering model – upload the pickle file and use pickle file name and model name when registering it**
12. **Ambari for big data logging**
13. **Submit a training run – custom role workspace/environment/write Actions element**
14. **Service Principal – Client Secret for authentication that will then give you a token**
15. **Service Principal – Automatic authentication. Grant access to your workspace and set up authentication.**
16. **For deployment - InferenceConfig gets the entry script with run and init as well as the Conda YML**
17. **To test or use the service: service\_url Authentication (InteractiveLoginAuthentication)**
18. **Authentication with keys (enabled by default for AKS and disabled by default for ACI. Token disabled by default for AKS, is not available for CI but set auth\_enabled=False)**
19. **Push notification on deployment =- > Event subsciption and web hook**
20. **Model.deploy gives you service - service.status, service.get\_logs() and service.scoring.uri. The scori\_uri is used as endpoint in request.post( endpoint…After Model deployment, all services can be found in ws.webservices**
21. You've deployed your web service as an Azure Container Instance (ACI) service that requires no authentication. This is fine for development and testing, but for production you should consider deploying to an Azure Kubernetes Service (AKS) cluster and enabling authentication. This would require REST requests to include an **Authorization** header.
22. Conda dependencies for Batch inferencing compute cluster
23. cd = CondaDependencies.create(pip\_packages=['scikit-learn','azureml-defaults','azureml-core','azureml-dataprep[fuse]'])
24. Conda dependencies for Batch inferencing
25. **In batch inferencing you use a ParallelRunConfig and a ParallelRunStep to create a Pipeline.** rest\_endpoint = published\_pipeline.endpoint (compare to service.scoring\_uri)
26. **In AutoMLConfig set model\_explainability=true to get feature importance**
27. When you've deployed a model into production as a service, you'll want to monitor it to track usage and explore the requests it processes. You can use Azure Application Insights to monitor activity for a model service endpoint.
28. *# Enable AppInsights*
29. aci\_service.update(enable\_app\_insights=**True**)
30. https://www.youtube.com/watch?v=qz1S-tv6iZs

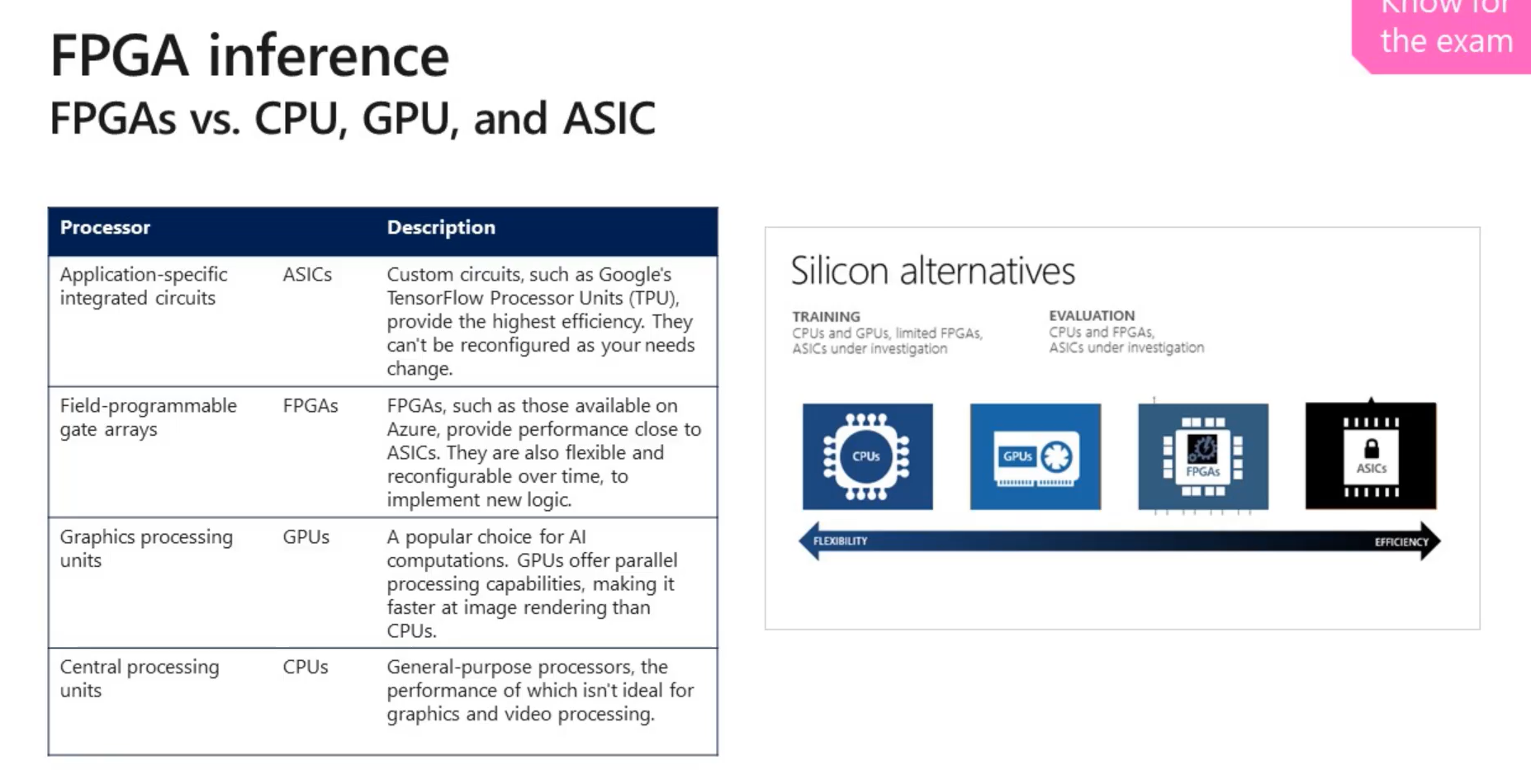
**Data drift withTimestmapColumn(‘date’) create new target dataset** DataDriftDetector.create\_from\_datasets. It tells you if it is time to retrain your model. It uses AI on your data to detect drift

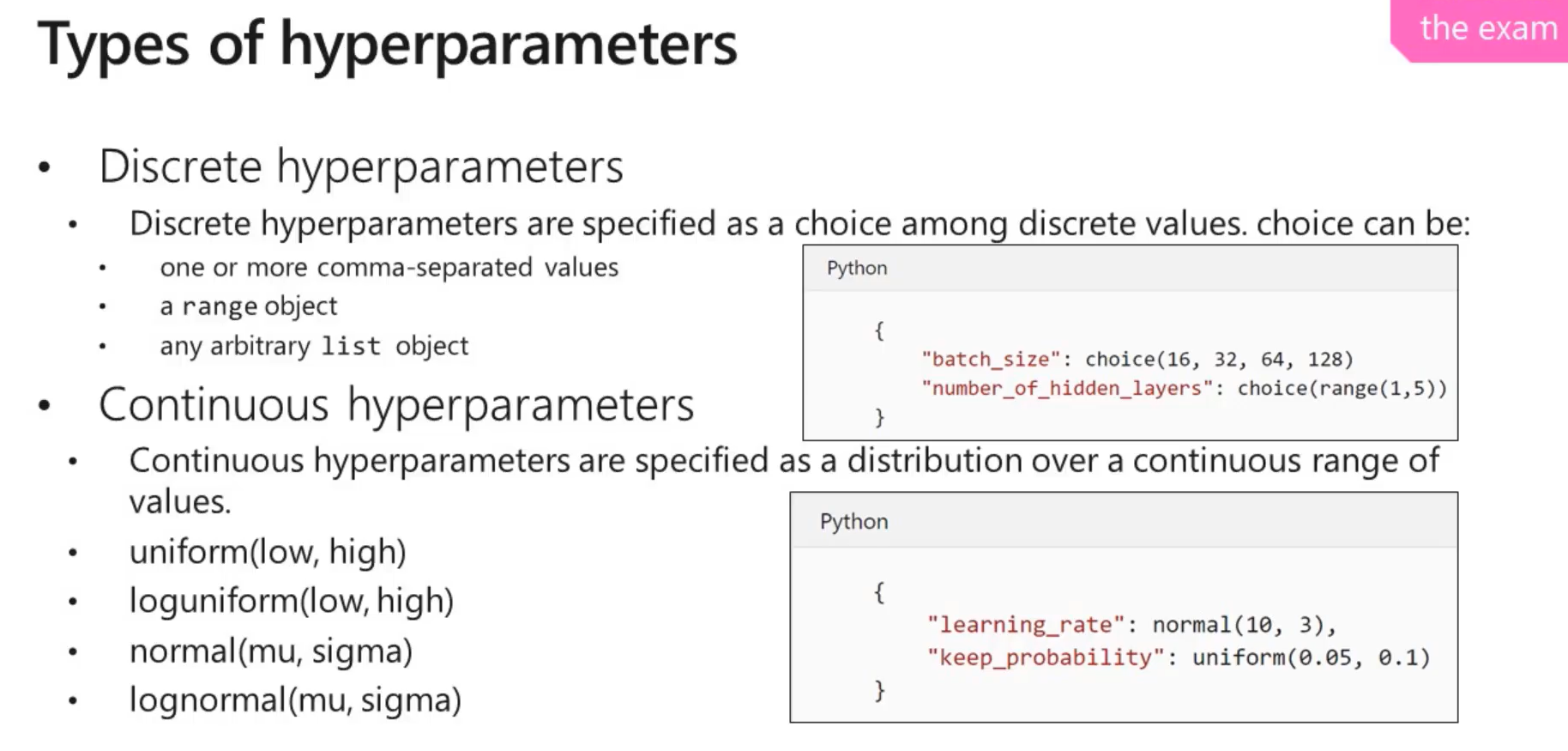
1. **Image classification and Designer in the Enterprise edition**
2. **Private\_endpoint\_auto – Not specifying this will ensure it is automatically approved**
3. **Service.run to run a deployed model**
4. **InferenceConfig object with the python scipt file (containing init and run) and an environment (with conda dependencies) is provided to deploy a model- model.deply()**

**NOTEBOOKS**

[**https://github.com/MicrosoftLearning/DP100/blob/master/07B%20-%20Creating%20a%20Batch%20Inferencing%20Service.ipynb**](https://github.com/MicrosoftLearning/DP100/blob/master/07B%20-%20Creating%20a%20Batch%20Inferencing%20Service.ipynb)

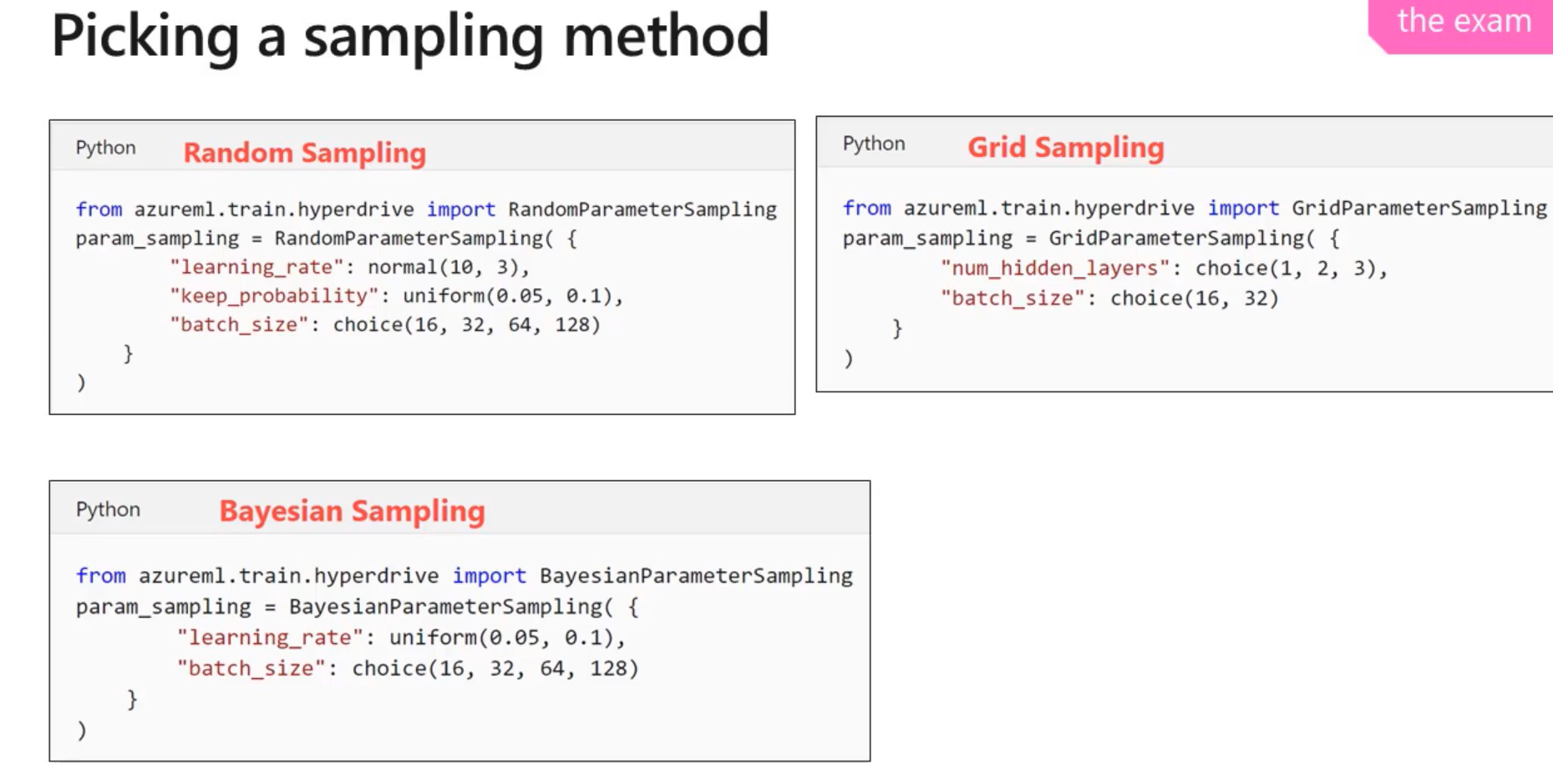
[**https://github.com/MicrosoftLearning/DP100/blob/master/07A%20-%20Creating%20a%20Real-time%20Inferencing%20Service.ipynb**](https://github.com/MicrosoftLearning/DP100/blob/master/07A%20-%20Creating%20a%20Real-time%20Inferencing%20Service.ipynb)

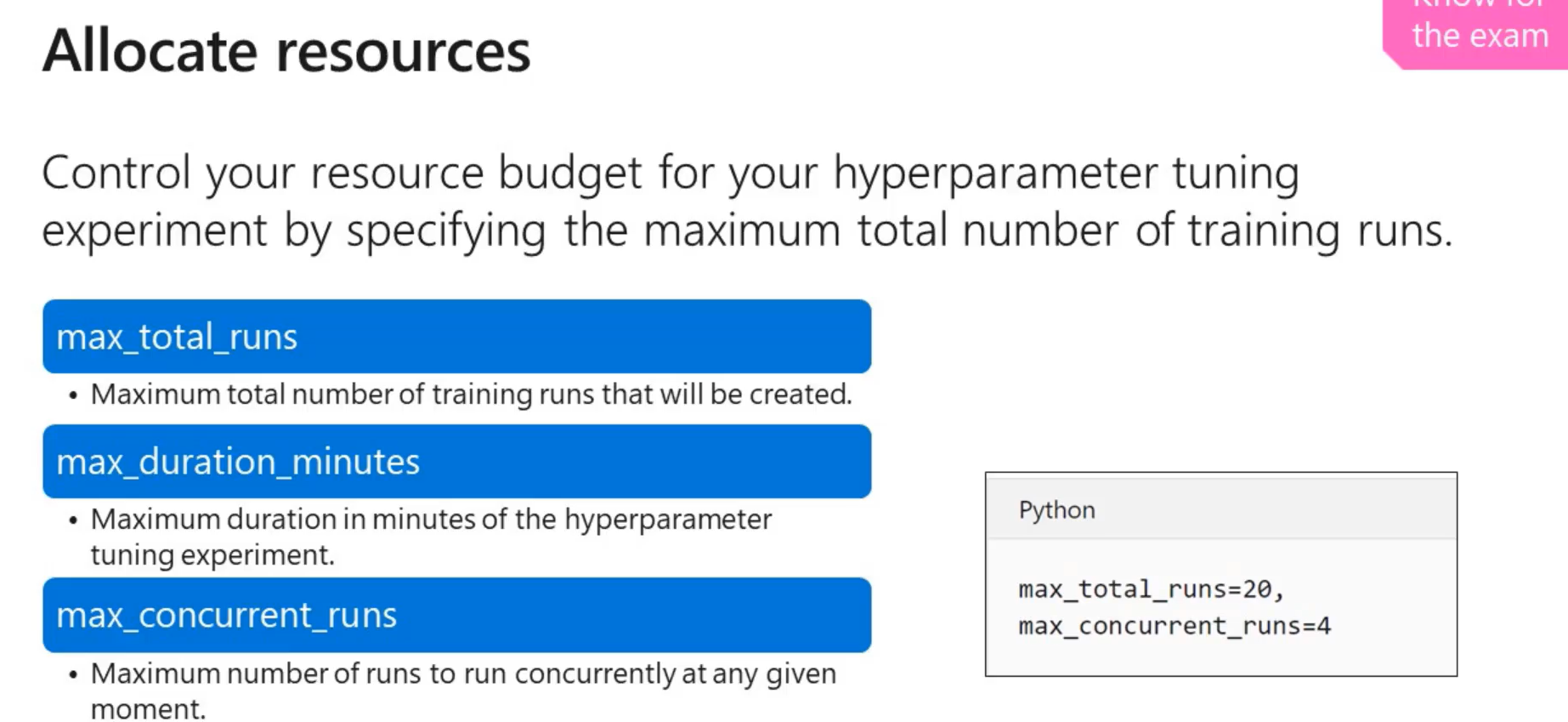






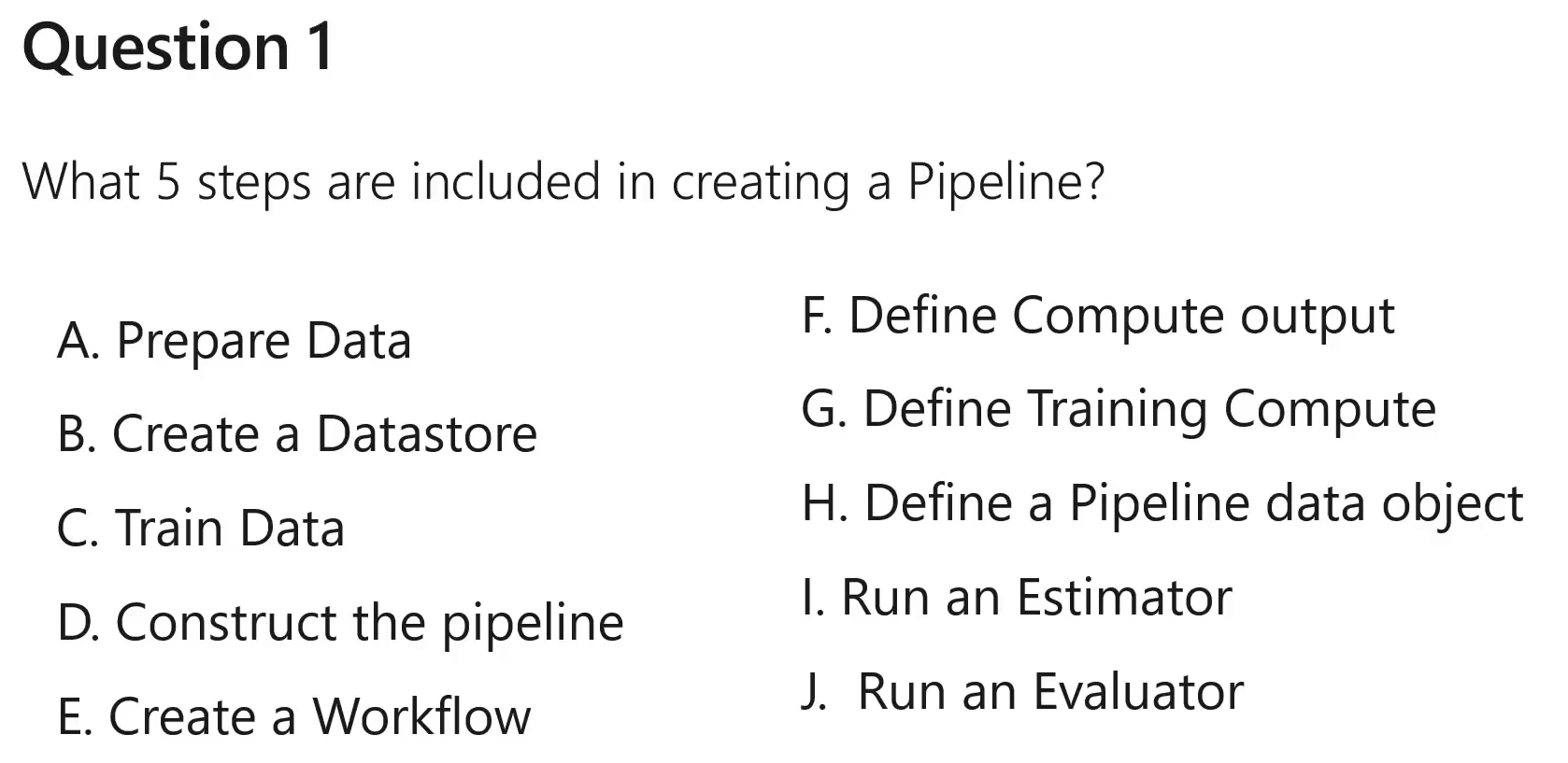




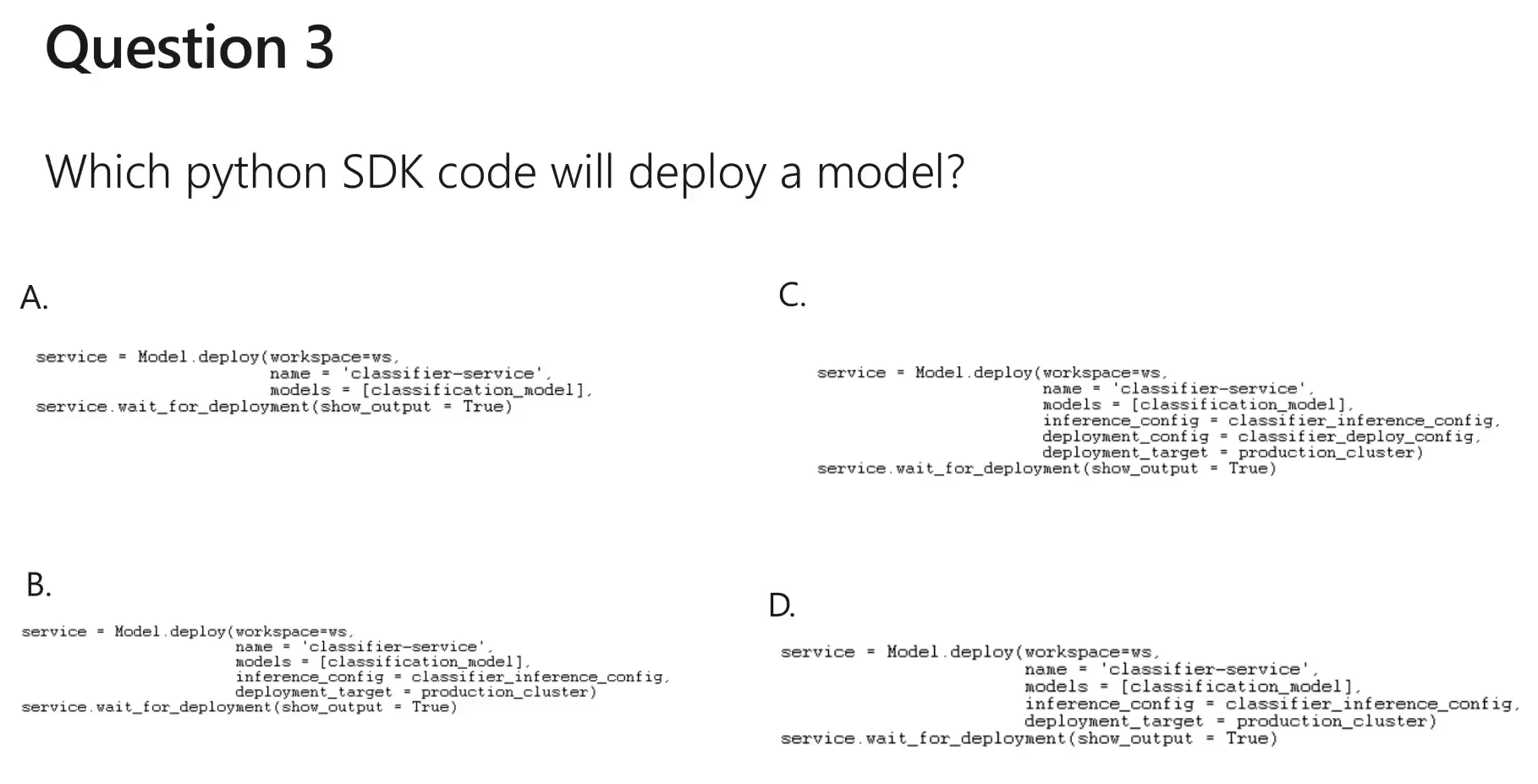




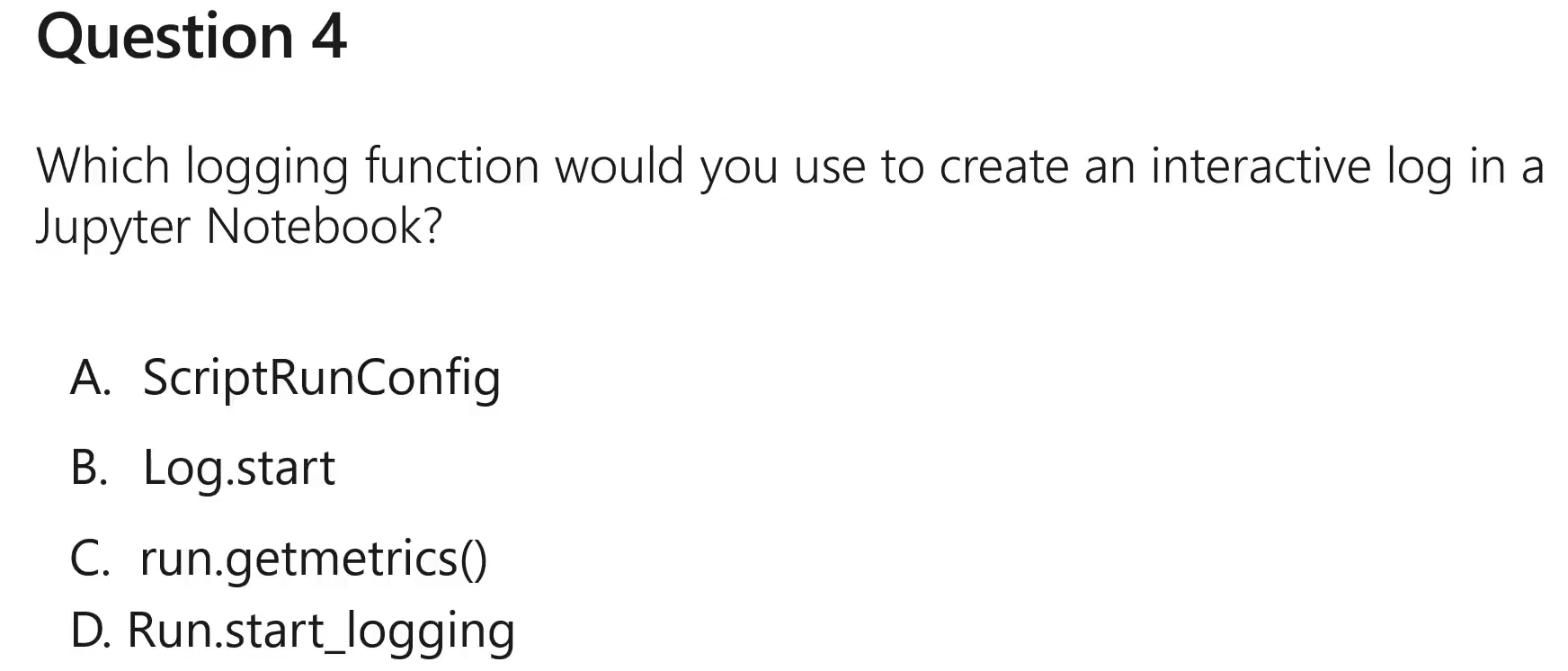
**PRACTICE QUESTIONS FROM WEBINAR**



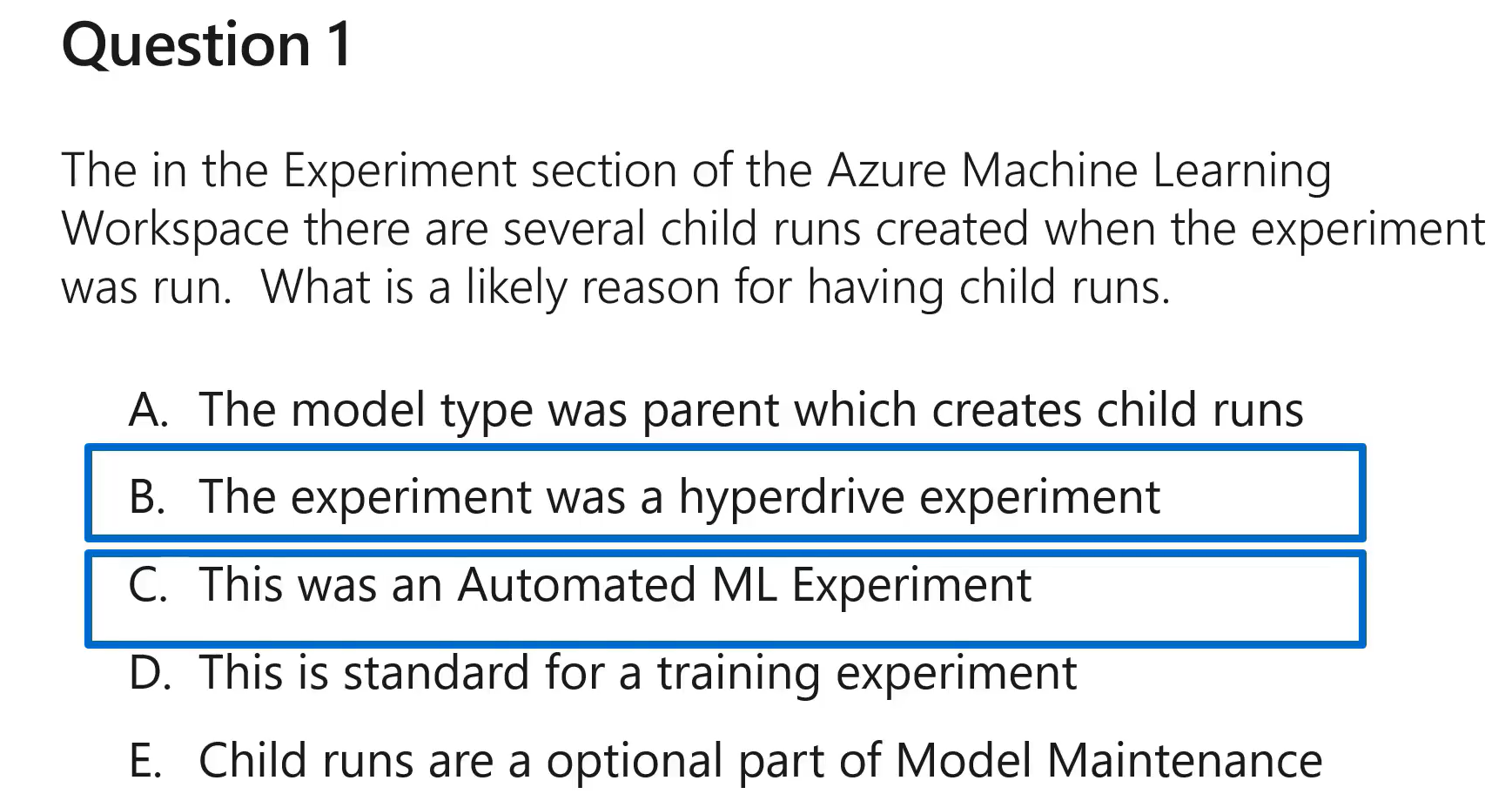


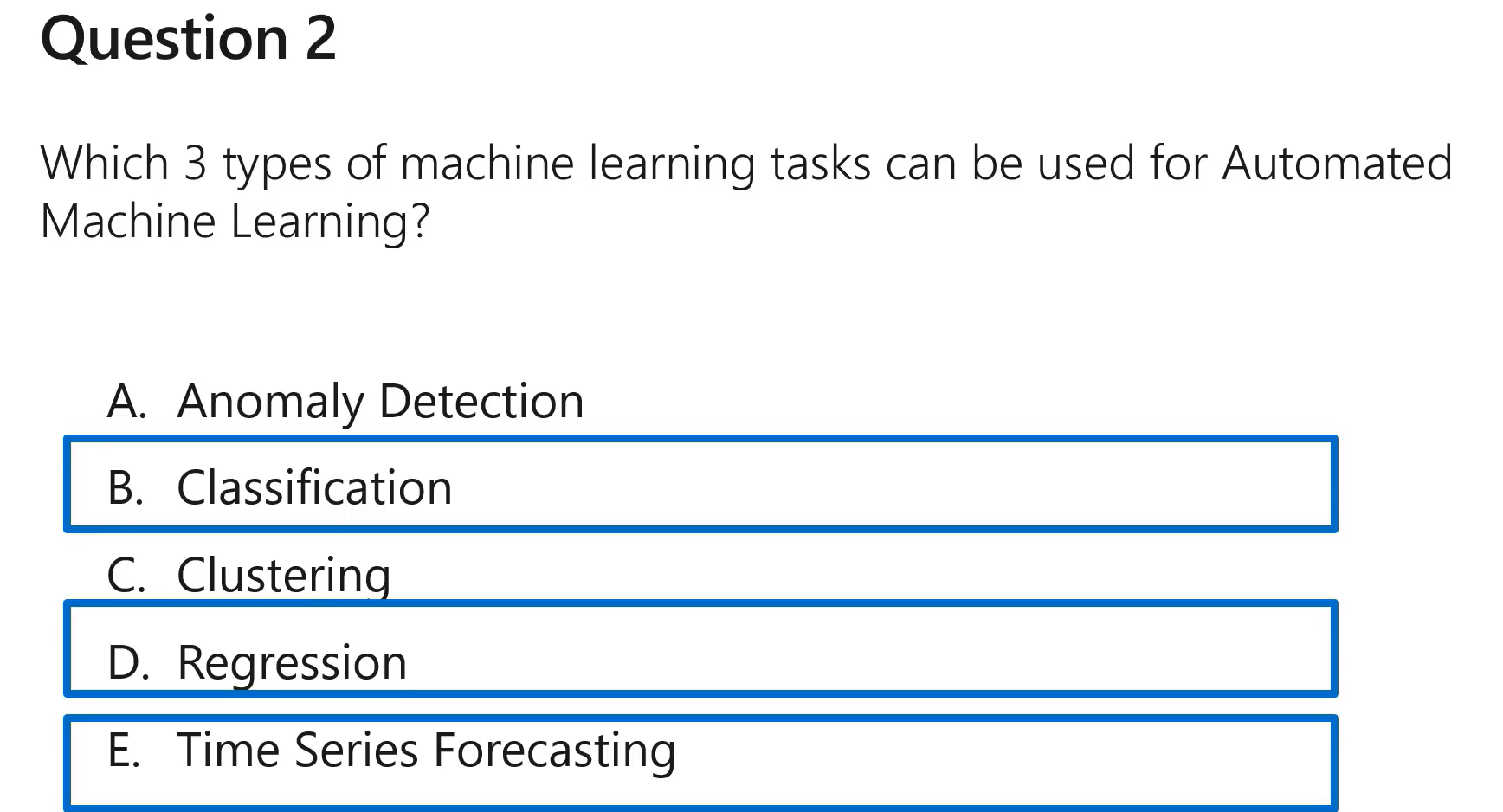
**`**

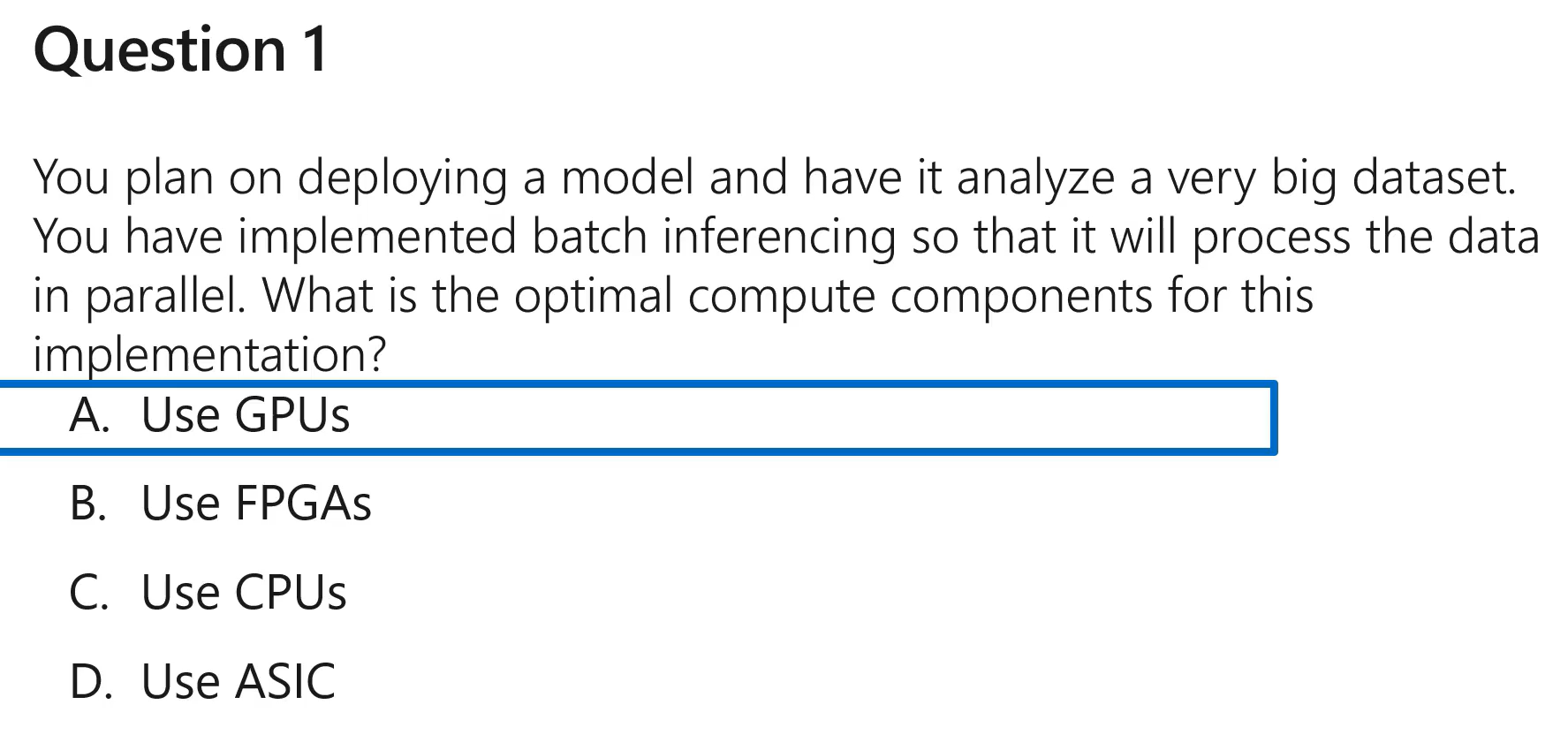


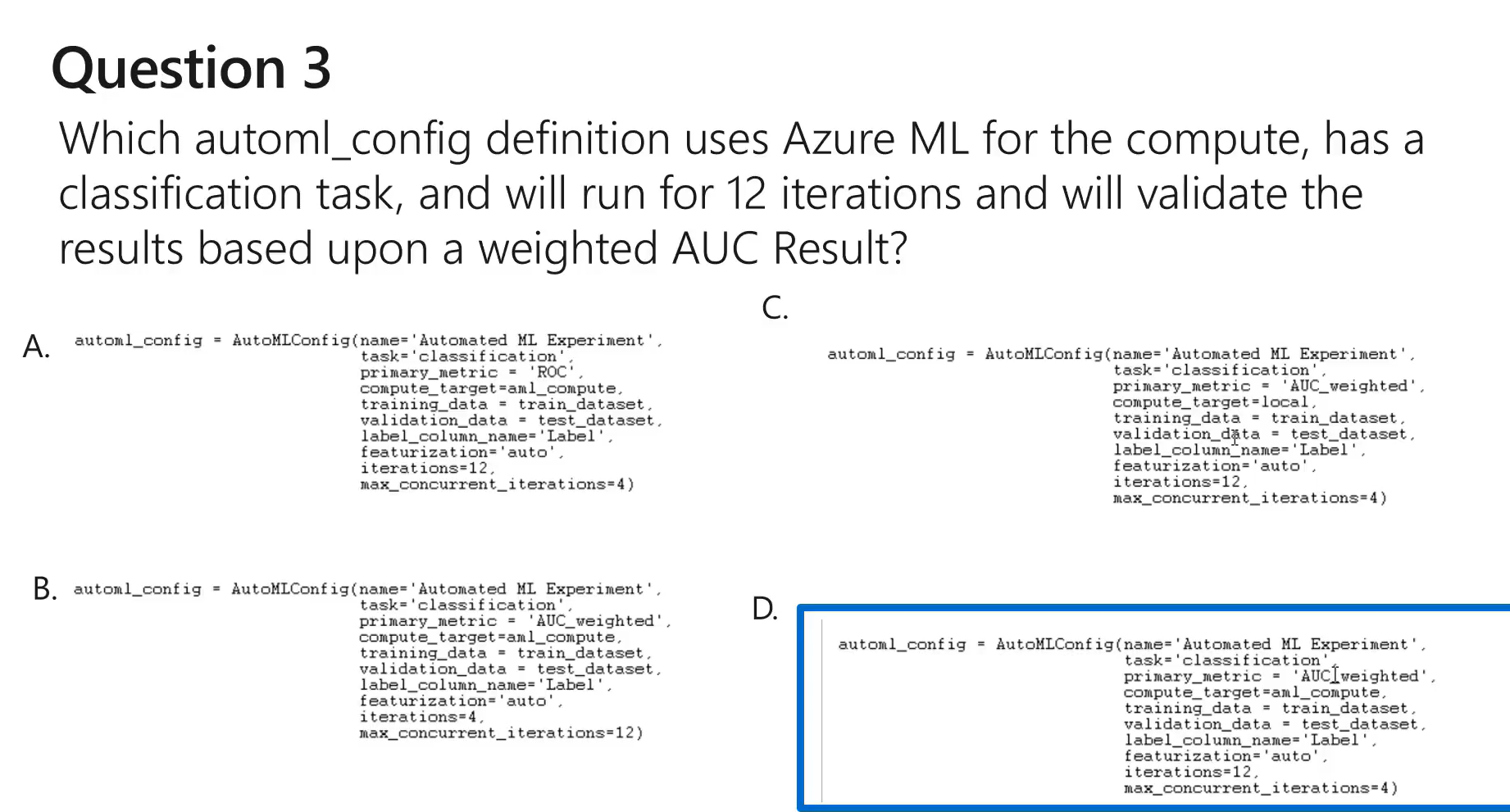


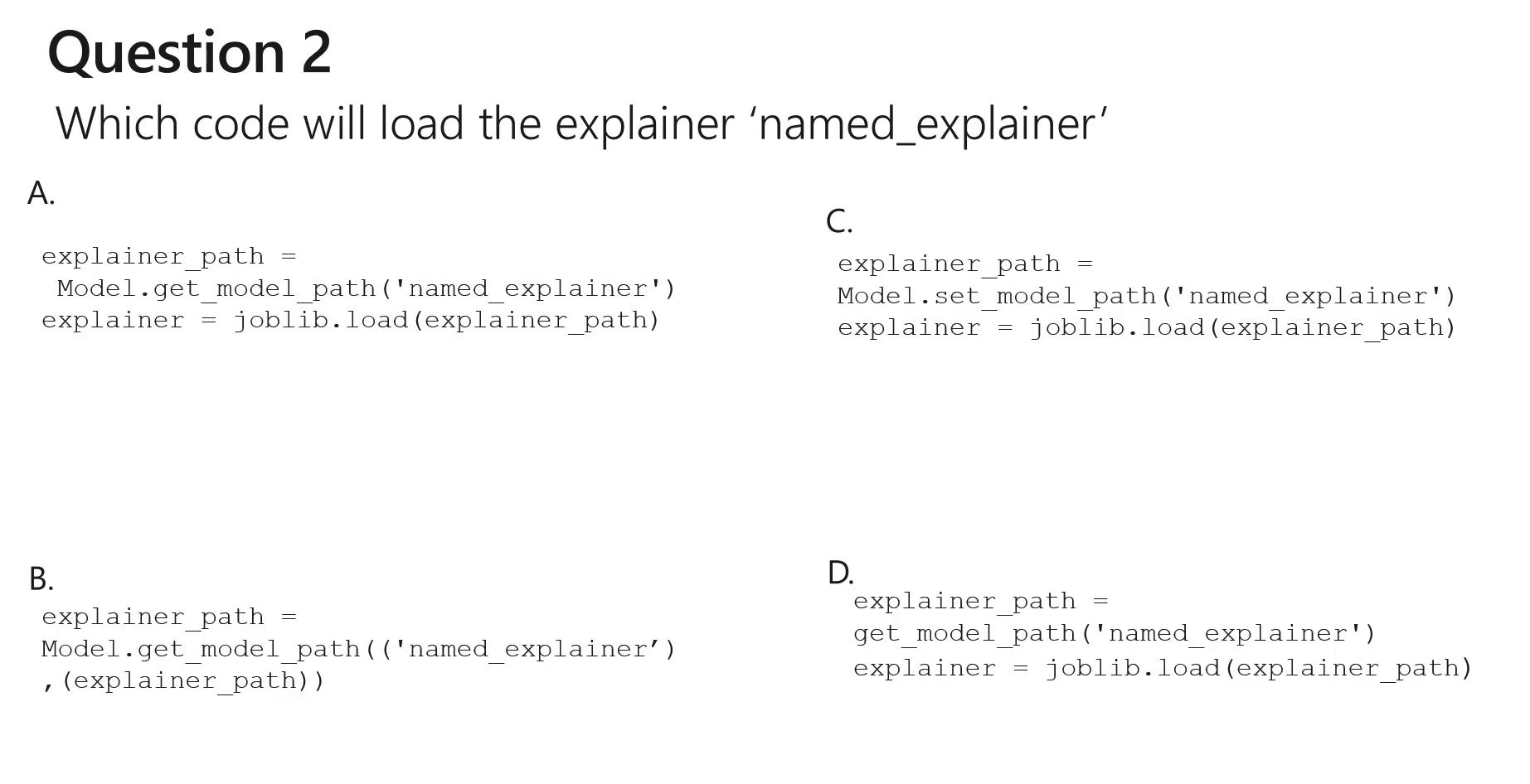


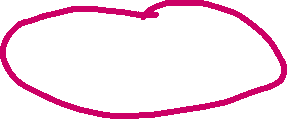


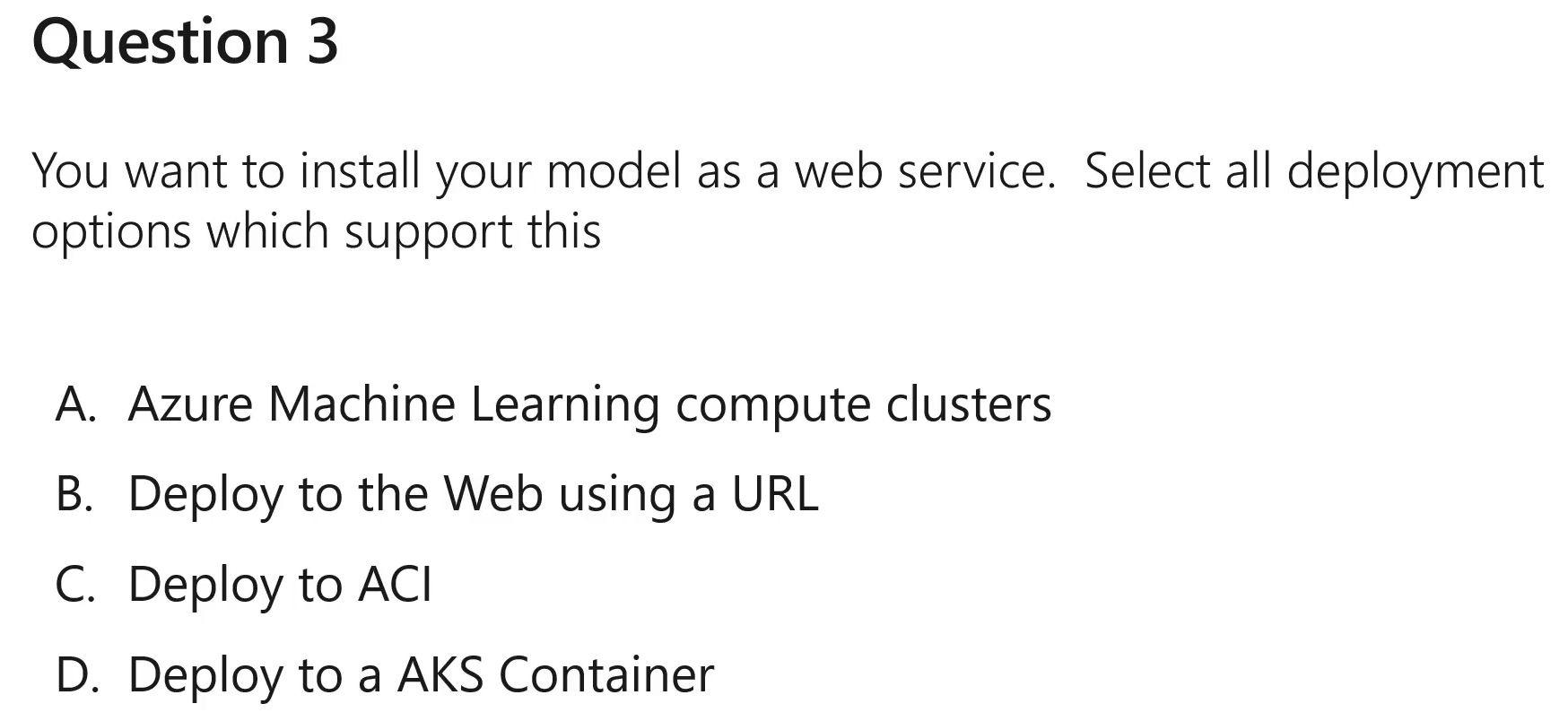




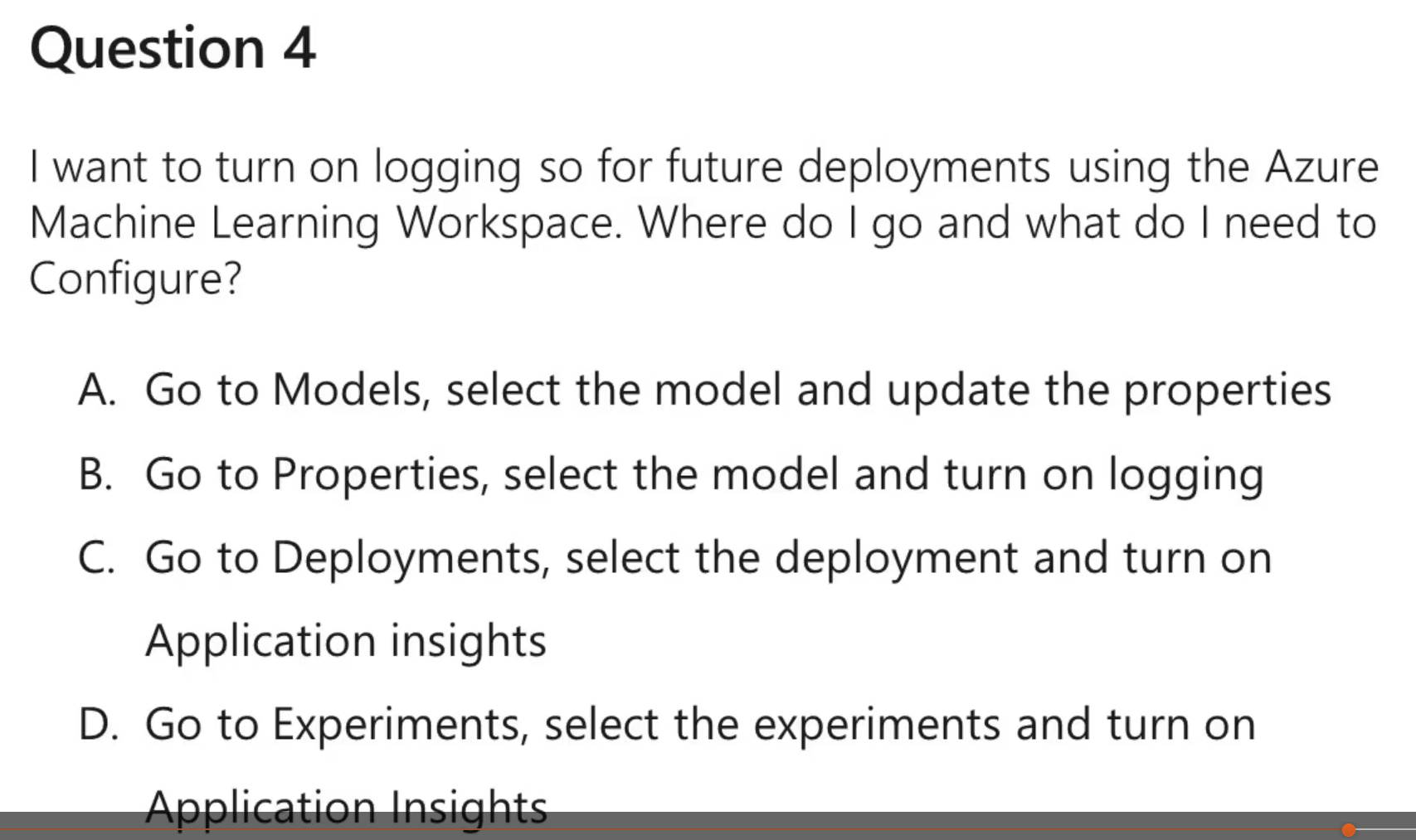




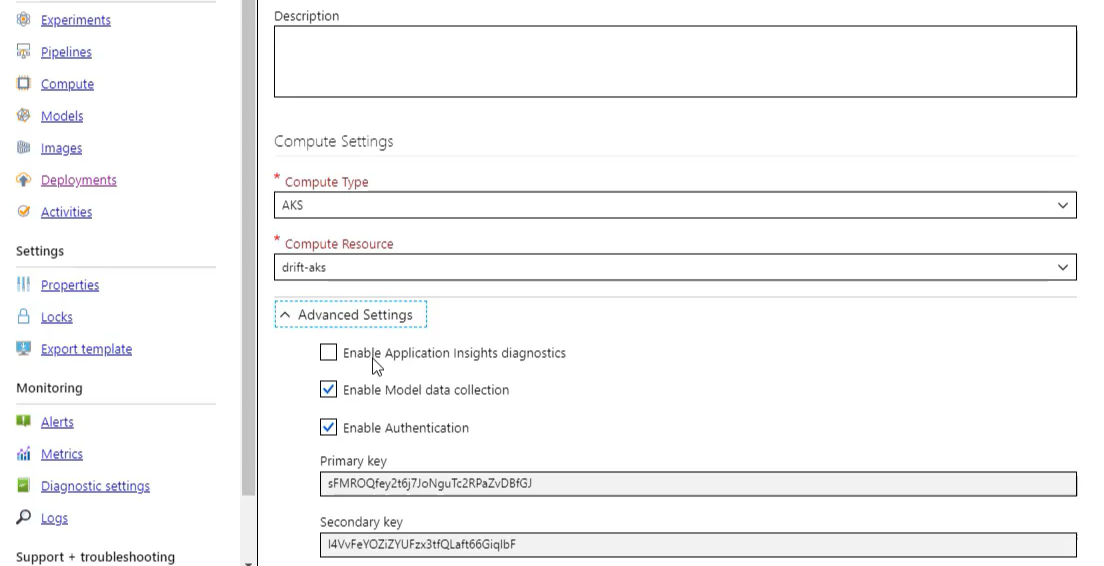


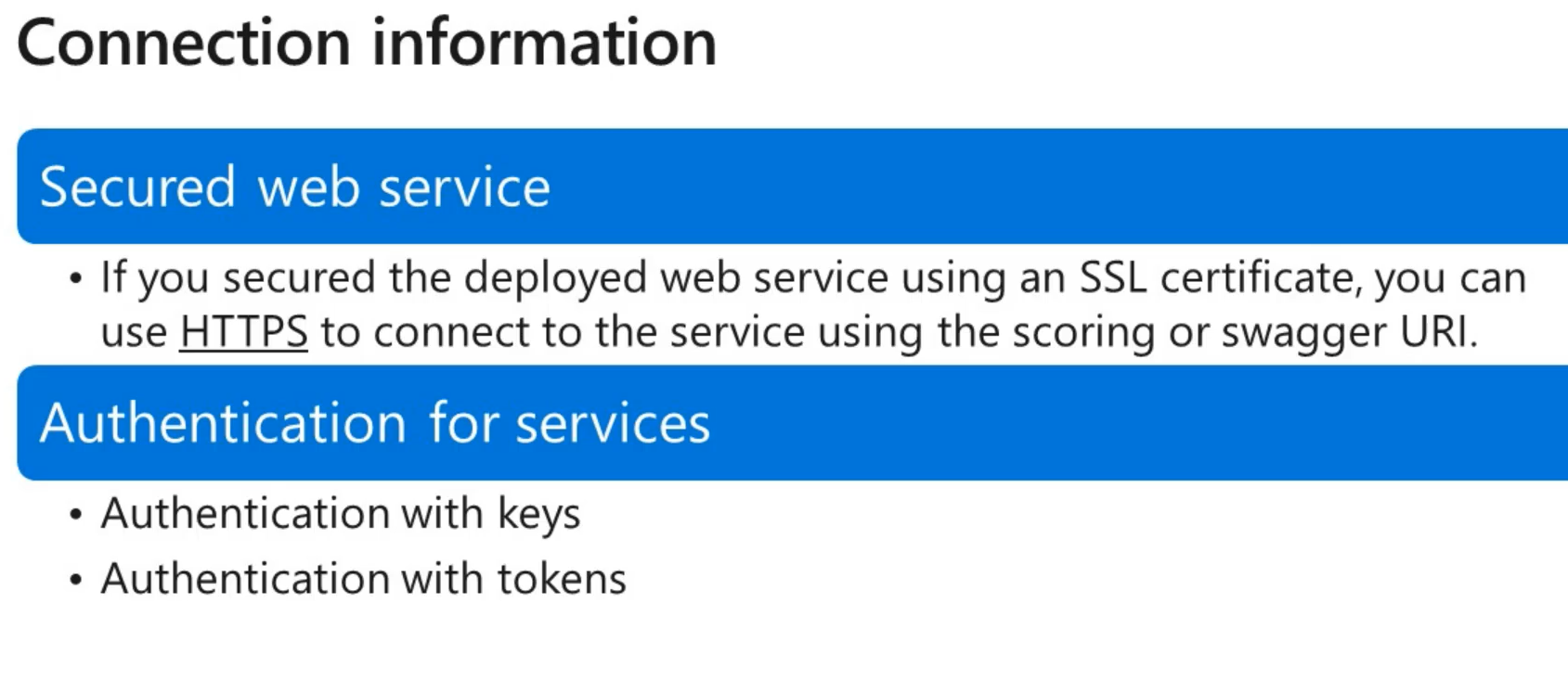












**Three steps for Batch inferencing**

1. **The Dataset is partitioned**
2. **Delegate the batch inferences to scaled compute**
3. **Output is stored in defined storage**